Appendix of python Scripts:

**1. Data\_flow.py**

There are 3 Functions in this script:

Dataflow:

Simulates flow of data for creating the csv file in GenTrainData.py

flowLive:

Manages the length of data arrays in Gen\_inputs.py

dataStream

Manages the length of data arrays in SVM\_live\_withdata.py

**2. TA\_indicators.py**

Contains functions for generating all the signals from the 12 technical indicators used.

**3. GenTrainData.py**

Generates the signals for every minute of the training data using the functions in TA\_indicators.py & writes these signals to a csv file for later use in training.

**4. TestProfit.py**

This Script:

* Trains Models using the csv file created in GenTrainData.py
* Saves Models for later use on live data
* Evaluates the profit and PV of the model over the training data.
* Creates graphs of profit as seen in the results section of the solution in the write up.

**5. Gen\_inputs.py**

Generates signals for the live data and outputs 1 vectors of signals. Equivalent to 1 entry in the csv file generated by functions in GenTrainData.py.

**6. client\_datastream.py**

Client listening on Bitmex web socket, downloads current data to keep a stream of the last n-minutes. This is needed as indicators need a buffer of n-minutes. Otherwise, it would take 1-2 hours after starting a model for it to begin trading as it would need to accumulate this buffer.

**7. SVM\_live\_withdata.py**

This Script:

* Loads Models for use on live data
* Utilizes the above scripts to create the input vector for the model & decides to go long/ go short
* Simulates trading based of these decisions
* Keeps track of current position, Profit & PV score
* Writes the models mistakes to a csv file for later use & improvement of future models
* Writes the models trades to a csv file for keeping track of the history of every trade a model has made. A history of the profit and PV value are also included in this csv file calculated after each trade.

**8. scikit.py**

Used to generate the graphs of accuracy scores of the ML algorithms, Confusion Matrices.